US006526556B1

NPO-20857-1 CU

(12) **United States Patent**
Stoica et al.

(10) **Patent No.:** US 6,526,556 B1
(45) **Date of Patent:** Feb. 25, 2003

(54) **EVOLUTIONARY TECHNIQUE FOR AUTOMATED SYNTHESIS OF ELECTRONIC CIRCUITS**

(75) Inventors: **Adrian Stoica**, Altadena, CA (US); **Carlos Harold Salazar-Lazaro**, Pasadena, CA (US)

(73) Assignee: **The United States of America as represented by the Administrator of the National Aeronautics and Space Administration**, Washington, DC (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 224 days.

(21) Appl. No.: **09/591,386**

(22) Filed: **Jun. 7, 2000**

Related U.S. Application Data

(63) Continuation-in-part of application No. 09/395,235, filed on Sep. 13, 1999.

(51) Int. Cl.^{7} **G06F 17/50**
(52) U.S. Cl. **716/16**; 716/1; 716/17; 706/13
(58) Field of Search 716/16, 17, 1; 706/13

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,867,397 A		2/1999	Koze et al.	364/489
5,897,628 A		4/1999	Kitano	706/13
5,963,975 A	*	10/1999	Boyle et al.	711/147
6,155,725 A	*	12/2000	Scepanovic et al.	395/500.1

OTHER PUBLICATIONS

Adrian Stoica, "Toward Evolvable Hardware Chips: Experiments with a Programmable Transistor Array," IEEE, Apr. 1999, pp. 1–7.*

Stoica et al, "Toward On–Board Synthesis and Adaptation of Electronic Functions: an Evolvable Hardware Approach," IEEE, Mar. 1999, pp. 351–357.*
Zebulum et al, "Evolvable Hardware: on the Automatic Synthesis of Analog Control Systems," IEEE, Mar. 2000, pp. 451–463.*
Augusto et al, "Analog Fault Diagnostic in Nonlinear DC Circuits with an Evolutionaly Algorithm," IEEE, Jul. 2000, pp. 609–616.*
Stoica et al, "Evolution of Analog Circuits on Field Programmable Transistor Arrays," IEEE, Jul. 2000, pp. 1–10.*
Adrian Stoica, "Evolvable Hardware: From on–Chip Circuit Synthesis to Evolvable Space Systems," IEEE, May 2000, pp. 1–9.*
Stoica et al, "Automatic Synthesis and Fault–Tolerant Experiments on an Evolvable Hardware Platform," IEEE, Oct. 2000, pp. 465–471.*
Stoica, "Reconfigurable Transistor Arrays for Evolvable Hardware," NASA Tech Brief, vol. 25, No. 2, Item # from JPL New Technology Report NPO–20078, Jul. 26, 1996, pp. i, 1, 1a–5a.*

(List continued on next page.)

Primary Examiner—Vuthe Siek
(74) *Attorney, Agent, or Firm*—John H. Kusmiss

(57) **ABSTRACT**

A method for evolving a circuit comprising configuring a plurality of transistors using a plurality of reconfigurable switches so that each of the plurality of transistors has a terminal coupled to a terminal of another of the plurality of transistors that is controllable by a single reconfigurable switch. The plurality of reconfigurable switches being controlled in response to a chromosome pattern. The plurality of reconfigurable switches may be controlled using an annealing function. As such, the plurality of reconfigurable switches may be controlled by selecting qualitative values for the plurality of reconfigurable switches in response to the chromosomal pattern, selecting initial quantitative values for the selected qualitative values, and morphing the initial quantitative values. Typically, subsequent quantitative values will be selected more divergent than the initial quantitative values. The morphing process may continue to partially or to completely polarize the quantitative values.

35 Claims, 4 Drawing Sheets
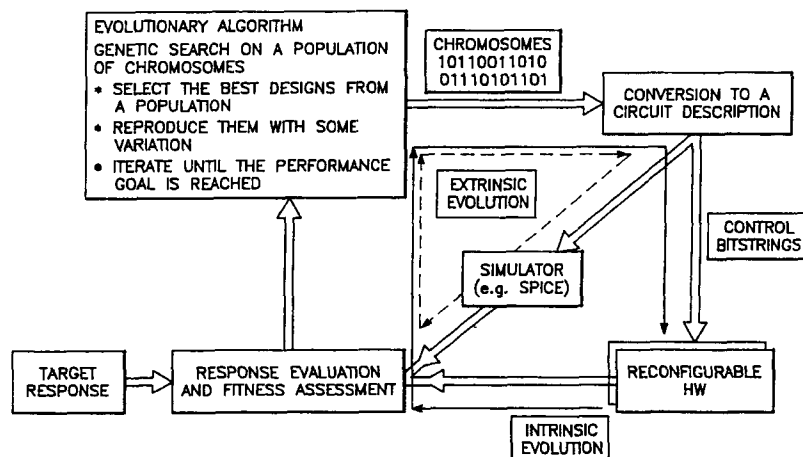
OTHER PUBLICATIONS

Bennett III, F., Andre, D., Koza, J.R., and Keane, M.A., "Evolution of a 60 decibel Op Amp using genetic programming," First International Conference, *ICES96*, Tsukuba, Japan, Oct. 7–8, 1996, Springer, pp. 455–469.

Flockton, Stuart, J., and Sheehan, Kevin, "Intrinsic Circuit Evolution Using Programmable Analogue Arrays," Second International Conference, *ICES98*, Lausanne, Switzerland, Sep. 23–25, 1998, Springer, pp. 144–153.

Iba, H., Iwata, M. and Higuchi, T., "Machine Learning Approach to Gate–Level Evolvable Hardware," First International Conference, *ICES96*, Tsukuba, Japan, Oct. 7–8, 1996, Springer, pp. 327–343.

Kajitani, I., Hoshino, T., Nishikawa, D., Yokoi, H., Nakaya, S., Yamauchi, T., Inuo, T., Kajihara, N., Iwata, M., Keymeulen, D., Higuchi, T., "A Gate–Level Ettwchip: Implementing GA operations and Reconfigurable Hardware on a Single LSI," Second International Conference, *ICES98*, Laussane, Switzerland, Sep. 23–25, Springer, pop. 1–12.

Koza, J.R., Andre, D., Bennett III, F., Keane, M.A., "Reuse Parameterized Reuse, and Hierarchical Reuse of Substructures in Evolving Electrical Circuits Using Genetic Programming, "First International Conference, *ICES96*, Tsukuba, Japan, Oct. 7–8, 1996, Springer, pp. 312–325.

Koza, J.R., Bennett III, F.H., Andre, D., Keane, M.A., "Automated WYWIWYG Design of Both the Topology and Component Values of Electrical Circuits Using Genetic Programming," *Proceedings of Genetic Programming Conference*, Stanford, 1996, pp. 28–31.

Koza, J.R., Dunlap, F., Bennett III, F., Keane, M.A., Lohn, J., and Andre, D., "Automated Synthesis of Computational Circuits Using genetic Programming," *IEEE Transactions on Evolutionary Computation*, vol. 1, No. 2, 1997, pp. 109–128.

Lohn, Jason D., and Colombano, S.P., "Automated Analog Circuit Synthesis Using a Linear Representation," Second International Conference, *ICES98*, Lausanne, Switzerland, Sep. 23–25, Springer, 1998, pp. 125–133.

Murakawa, M., Yoshizawa, S., Adachi, T., Suzuki, S., Takasuka, K., Iwata, M., Higuchi, T., "Analogue EHW Chip for Intermediate Frequency Filters," Second International Conference, *ICES98*, Lausanne, Switzerland, Sep. 23–25, 1998, Springer, pp. 132–143.

Stoica, A., "On Hardware Evolvability and Levels of Granularity," *International Conference on Intelligent Systems and Semiotics*, NIST Gaithersburg, VA, Sep. 1997, pp. 1–4.

Thompson, Adrian, "Silicon Evolution," *Proceedings of Genetic Programming, MIT Press*, 1996, pp. 75–90.

Thompson, Adrian, "On the Automatic design of Robyst Electronics Through Artificial Evolution," Second International Conference, *ICES98*, Lausanne, Switzerland, Sep. 23–25, Springer, 1998, pp. 13–24.

Thompson, Adrian, "An Evolved Circuit, Intrinsic in Silicon, Entwined with Physics," First International Conference, *ICES96*, Tsukuba, Japan, Oct. 7–8, Springer, 1996, pp. 390–405.

Zebulum, R.S., Pacheco, M.A., Vellasco, M., "Evolvable Systems in Hardware design: Taxonomy, Survey and Applications," First International Conference, *ICES96*, Tsukuba, Japan, Oct. 7–8, Springer, 1996, pp. 344–358.

Zebulum, R.S., Pacheco. Marco, A., Vellasco, M., "Analog Circuits Evolution in Extrinsic and Intrinsic Modes," Second International Conference, *ICES98*, Lausanne, Switzerland, Sep. 23–25, Springer, 1998, pp. 154–165.
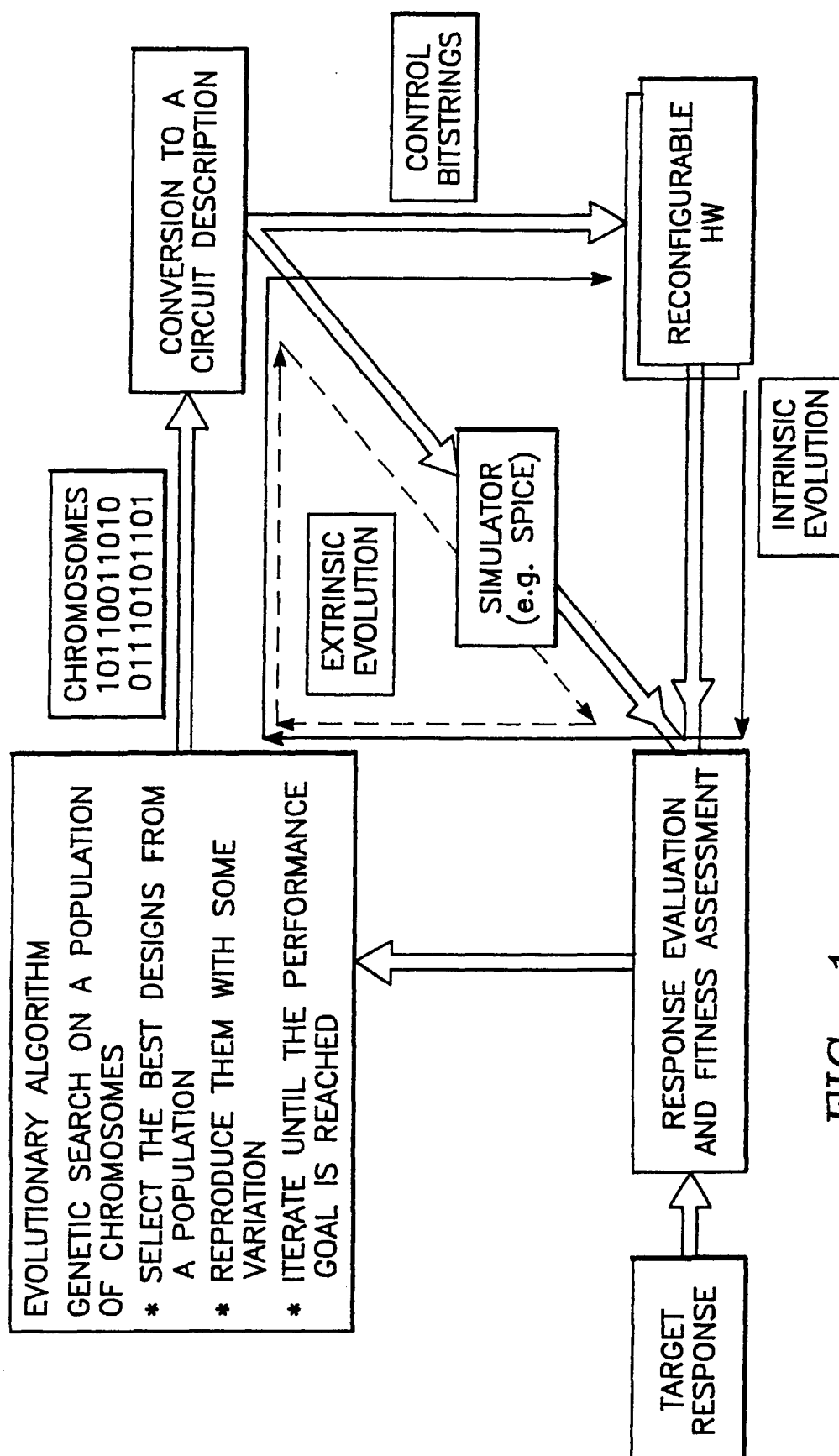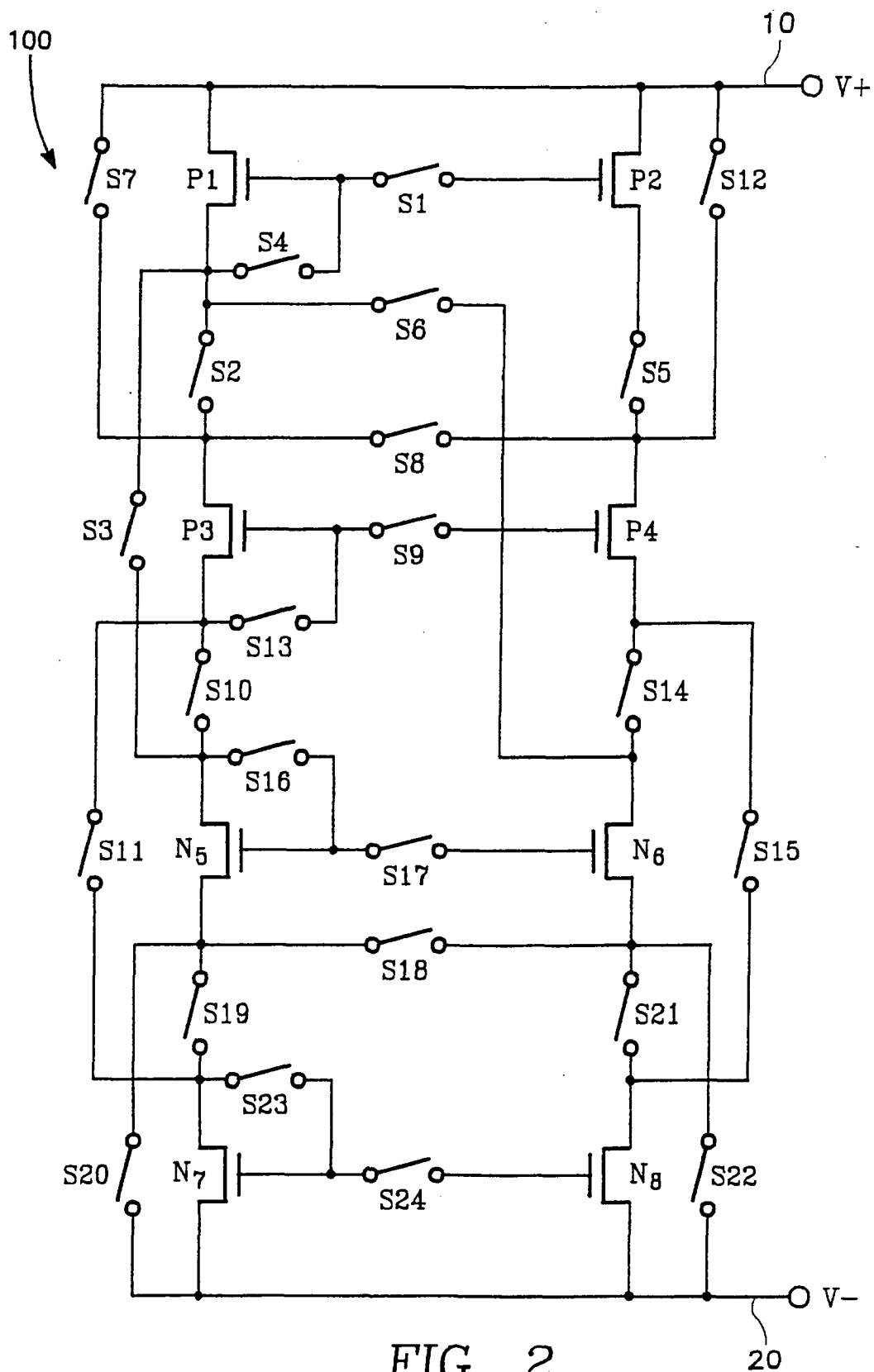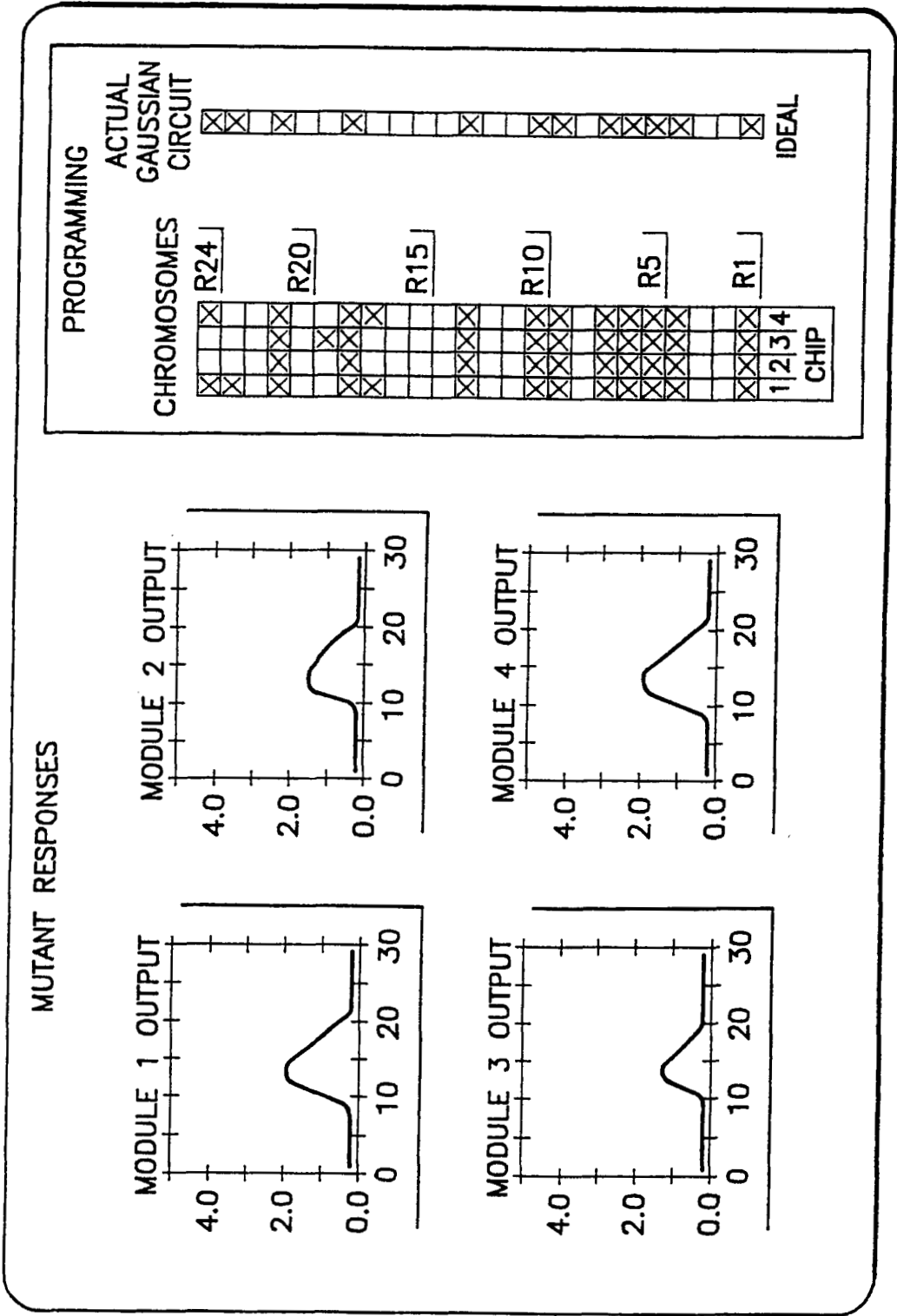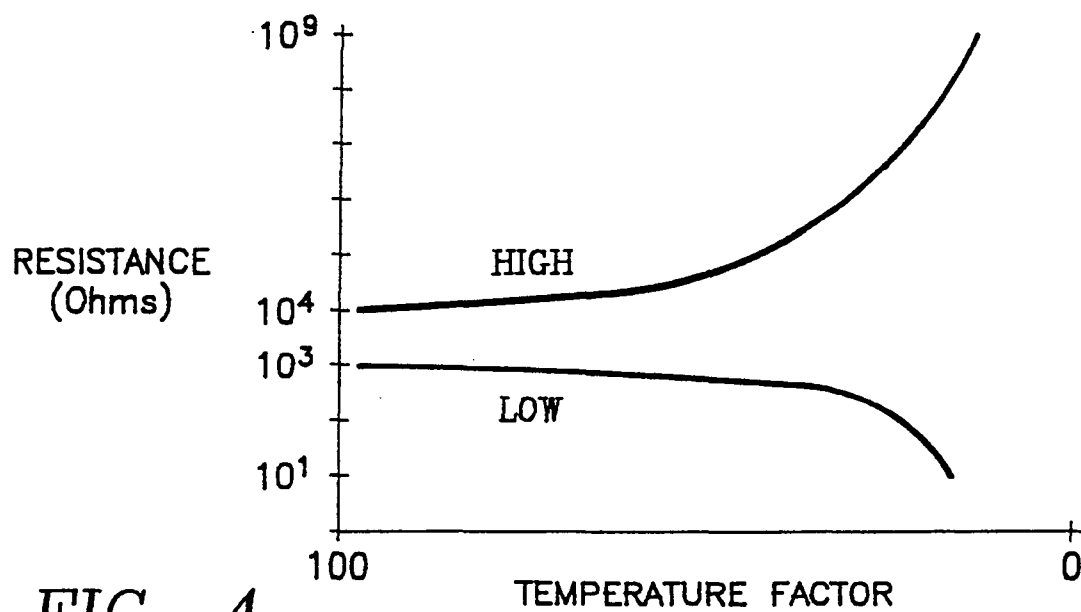
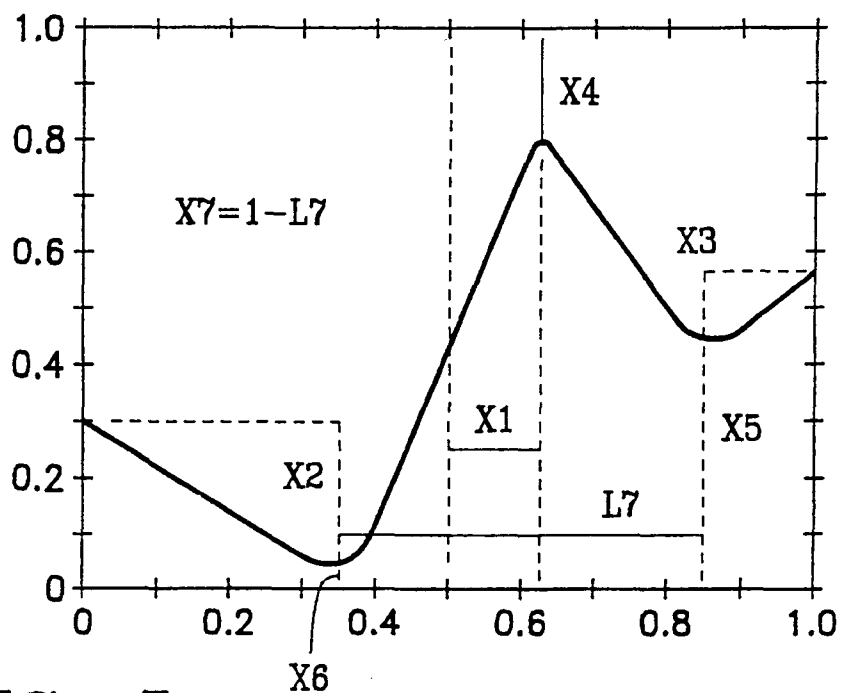* cited by examiner

FIG. 1

FIG. 2

FIG. 3

FIG. 4

FIG. 5

1

EVOLUTIONARY TECHNIQUE FOR AUTOMATED SYNTHESIS OF ELECTRONIC CIRCUITS

CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation-in-part of U.S. application Ser. No. 09/395,235, by Adrian Stoica and Carlos Salazar-Lazaro, filed on Sep. 13, 1999, entitled EVOLU-TIONARY TECHNIQUE FOR AUTOMATED SYNTHE-SIS OF ELECTRONIC CIRCUITS, which is herein incorporated by reference in its entirety.

BACKGROUND

Evolvable Hardware or EHW is reconfigurable hardware that self-configures under the control of an evolutionary algorithm. In evolutionary electronics, the search for an electronic circuit reconfiguration can be made in software and the final solution downloaded or become a blueprint for hardware, which is referred to as extrinsic evolution. Alternatively, evolution can be performed directly in hardware, referred to as intrinsic evolution. With intrinsic evolution, solutions may be evolved directly on a chip.

The main steps of evolutionary synthesis are illustrated in FIG. 1. First, a population of chromosomes is randomly generated. The chromosomes are converted into circuit models for extrinsic EHW, or control bit strings downloaded to programmable hardware for intrinsic EHW. Circuit responses are compared against specifications of a target response, and individuals are ranked based on how close they come to satisfying it. In preparation for a new iteration loop, a new population of individuals is generated from the pool of best individuals in the previous generation, some of these individuals are taken as they were and some are modified by genetic operators such as chromosome cross-over and mutation. This process is repeated for many generations, and results in increasingly better individuals. Such a process is usually stopped after a number of generations, or when the closeness to the target response has reached a sufficient degree. One of several solutions may be found among the individuals of the last generation.

A variety of circuits have been synthesized through extrinsic evolutionary means. For example, Koza et al., in U.S. Pat. No. 5,867,397, issued on Feb. 2, 1999, entitled METHOD AND APPARATUS FOR AUTOMATED DESIGN OF COMPLEX STRUCTURES USING GENETIC PROGRAMMING, herein incorporated by reference in its entirety, used Genetic Programming (GP) to grow an "embryonic" circuit to a circuit that satisfies desired requirements. This approach was used for evolving a variety of circuits, including filters and computational circuits. An alternative encoding technique using a linear representation, which has the advantage of reduced computational load, has been used in for automated filter design.

With these extrinsic approaches, though, evolutions of analog circuits were performed in simulations, without concern of a physical implementation, but rather, as a proof-of-concept that evolution can lead to designs that compete or even exceed the performance of human designs. Although in principle, one can test their validity in circuits built from discrete components, or in an ASIC, no analog programmable devices exist that would support the implementation of the resulting design. Thus, these approaches do not provide a practical solution to intrinsic evolution.

Intrinsic evolution can speed-up the search for a solution circuit by a few orders of magnitude compared to evolution

2

in software simulations, specifically if one simulates large, complex analog circuits, and if the circuit response is rapid. Moreover, since the software simulation relies on models of physical hardware with limited accuracy, a solution evolved in software may behave differently when downloaded in programmable hardware; such mismatches are avoided when evolution takes place directly in hardware. Further, unlike software evaluation where more complex circuitry and more accurate modeling takes longer to evaluate, hardware evolution scales well with both size of the circuits and model accuracy, thus providing less significant increases in evaluation time.

Although reconfigurable devices exist, they have a limited range of possible applications. On-chip evolution was demonstrated by A. Thompson, in *Silicon Evolution*, in Proceedings of Genetic Programming 1996) (GP96) , MIT Press, herein incorporated by reference in its entirety, using a Field Programmable Gate Array or FPGA as a programmable digital device, and a Genetic Algorithm or GA as the evolutionary mechanism.

Such a technique using gate arrays, however, is not practical for analog circuit evolution. Logical gates are not good elementary building blocks for analog circuits as they are designed optimized for logical/binary behaviors. For example, transistor interconnections that are designed to facilitate digital logic signals and flows do not necessarily provide good analog response and signal flow. The usage of the gate array for evolution can result in exploitation by evolution, of parasitic and unintended signal paths and functioning modes for the components. As a result, circuits may evolve in one region of a chip that can not be replicated in other parts, or on other chips, although the same genetic code is used.

Moreover, conventional on chip evolution has not provided sufficient granularity for practical applications. While several levels of granularity are in use, the most common digital devices are configurable at the gate-level. In the analog programmable devices, such as in Field Programmable Analog Arrays or FPAAs, the reconfigurable active elements are Operational Amplifiers, which have only very coarse granularity and little functionality with good precision, thus having only a limited range of possible applications.

Computation by analog circuitry has been lost as a technique for information processing because analog circuits were not easily programmable, and required precise components with no drifts. If the evolutionary mechanism and process proves sufficiently powerful for evolving complex analog circuits, then its combination with reconfigurable analog devices potentially will be able to capture the benefits of analog in new applications. As such, the potential of analog processing is much greater than what is able to be exploited today.

Analog circuitry has advantages in cost, size and power consumption (as compared to digital circuitry) and can directly process signals that are continual in time and amplitude. Even a single transistor has many functions such as generation of square, square-root, exponential and logarithmic functions, voltage-controlled current sources; analog multiplication of voltages, and short term and long term analog storage. As such, the basic combinations of transistors offer a rich repertoire of linear and nonlinear operators available for local and collective analog processing. Using evolution, the benefits of analog processing can be exploited, while its disadvantages reduced or even eliminated.

Also, it has been recognized by the inventors herein that evolutionary searches may perform significantly better with analog than with digital circuitry. A possible explanation lies in the fact that analog behaviors have relatively smoother spaces, which is better for the evolutionary search. Thus, new perspectives are possible: evolutionary searches offering automatic programming; sufficiently precise equivalent components could be obtained if the programmable analog components offer controllably of their operating points; and drifts that can be compensated for by adjusting operating points or, if the drifts are too strong, by a new search for a different optimal circuit configuration and operating point. Moreover, analog computation on simple low-power circuits can boost emerging applications areas of "smart matter" and distributed high bandwidth adaptive sensing.

Furthermore, a hardware implementation al.so offers a big advantage in evaluation time for a circuit; the time for evaluation is determined by the goal function. For example, considering an A/D converter operating at a 100 kHz sampling rate the electronic response of the A/D converter is available within 10 microseconds, compared to 1 second on a computer running SPICE; this advantage increases with the complexity of the circuits. In this case, the 10 speedup would allow evaluations of populations of millions of individuals in seconds instead of days. Moreover, the higher the frequency at which a circuit needs to function, the shorter is its evaluation time, making the design of very high frequency circuits an excellent candidate for intrinsic evolutionary design.

Thus, a practical solution to intrinsic evolution for programming analog devices is desirable. Furthermore, a shift in the design approach, from reconfigurable devices, to evolution-oriented devices or evolvable devices would facilitate hardware evolvability.

SUMMARY

A method for evolving a circuit comprising configuring a plurality of transistors using a plurality of reconfigurable switches so that each of the plurality of transistors has a terminal coupled to a terminal of another of the plurality of transistors that is controllable by a single reconfigurable switch. The plurality of reconfigurable switches are controlled in response to a chromosome pattern.

With this method, the plurality of reconfigurable switches may be controlled using an annealing function. As such, the plurality of reconfigurable switches may be controlled by selecting qualitative values for the plurality of reconfigurable switches in response to the chromosomal pattern, selecting initial quantitative values for the selected qualitative values, and morphing the initial quantitative values. Typically, subsequent quantitative values will be selected more divergent than the initial quantitative values. The morphing process may continue to partially or completely polarize the quantitative values.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 shows a simplified functional block diagram of evolutionary processes.

FIG. 2 is a preferred embodiment of a circuit in accordance with the present invention.

FIG. 3 illustrates several additional circuits obtained by evolution showing less than ideal but possibly useful circuit responses.

FIG. 4 represents the annealing type process used in some embodiments employing gradual switch states to provide evolution by morphing switch states.

FIG. 5 shows parameter used for specification of a fitness function.

DESCRIPTION OF THE PREFERRED EMBODIMENTS AND METHODS OF THE PRESENT INVENTION

The preferred embodiment of the present invention provides a circuit adapted for use in intrinsic evolution. The preferred embodiment provides a programmable transistor array or PTA, which is programmable at the transistor level. In a preferred embodiment of the present invention, transistors form a plurality of modules, which may be interconnected to other modules, or to other transistors of the PTA, to provide further functionality.

Turning to FIG. 2, the PTA module 100 is an array of transistors interconnected by programmable switches. These switches provided transistor terminal to transistor terminal, or transistor terminal to power terminal connections. The status of the switches (On or Off) determines a circuit topology and consequently a specific response. Thus, the topology can be considered as a function of switch states, and can be represented by a binary sequence, such as a chromosome pattern of "1011 . . . ", where by convention, one can assign 1 to a switch turned On and 0 to a switch turned Off. Programming the switches On and Off determines a circuit and provides a means responsive to a chromosome pattern for selectively controlling the coupling of the interconnections of the transistor array.

In the preferred embodiment, the PTA allows programming of both analog and digital circuits by intrinsic evolution. In a preferred embodiment, CMOS field effect transistors are used as the elemental building blocks, although other embodiments may employ other type devices, such as bipolar devices, single electron devices, quantum dot devices, resonant tunneling devices, optically coupled devices, or other similar devices, as elemental building blocks. CMOS transistors allow evolution to take advantage of inherent resistance and capacitance functions, as well as analog and digital transistor operation.

As all integrated circuits ultimately rely on functions implemented with transistors, the PTA of the preferred embodiment provides a versatile platform for synthesis of analog, digital, or even mixed-signal circuits using the same transistors. It also provides a more suitable platform for synthesis of analog circuitry than other reconfigurable devices and allows transferrable analog circuits to be evolved directly on the chip.

The optimal choice of elementary block type and granularity is task dependent. The preferred embodiment of the present invention provides reconfigurable modules 100 based on elements of the lowest level of granularity. With such a configuration, virtual higher-level building blocks can be implemented by imposing programming constraints. An example would be forcing groups of elementary cells to act as a whole, such as by freezing certain parts of the switch configuration bit string to provide digital gates or basic analog circuits.

Thus, the PTA of the preferred embodiment of the present invention may provide versatile functional cells to provide a higher level of functionality. More specifically, for example, certain parts of the switch configuration bit string could be frozen to provide interconnections between the N-type transistors to form a virtual NAND gate. Ideally, the virtual blocks for evolution should be automatically defined/clustered during evolution. This concept is analogous to the Automatically Defined Functions predicted and observed in software evolution.

As such, granularity selection allows for mixed granularity solutions. Higher level functionality may be combined with elemental functionality within modules on the same chip to allow evolved circuit designs not obtainable with only higher level cells. Also, such granularity selection allows for hybrid digital/analog solutions not readily obtainable in current arrays.

FIG. 2 illustrates an embodiment of a PTA module **100** having 8 transistors and 24 programmable switches. In this embodiment, the transistors P1–P4 are PMOS and N5–N8 are NMOS. The plurality of transistors P1–N8 are coupled between a power source terminal **10** and a power sink terminal **20** in a topology capable of admitting power between the power source terminal **10** and the power sink terminal **20**. In the embodiment of FIG. 2, the plurality of transistors P1–N8 are coupled via reconfigurable switches S1–S24 so that every transistor terminal to transistor terminal interconnection is via a reconfigurable switch.

Although it is preferred to provide a reconfigurable switch between every terminal to terminal interconnection, it is possible, in some embodiments, to provide a hard connection between certain transistor terminals, such as a hard bias connection, or a gate to gate connection. Although such hard connections will limit evolutionary circuit possibilities, such a connection may be desirable, for example, when evolution will be employed to evolve a specific family of circuits employing recurrent, frequent, or common connection types, or in other circumstances.

Thus, with the preferred embodiment, the PTA has at least one contiguous region within the array where every transistor terminal to transistor terminal connection is via a reconfigurable switch. Further, in such an embodiment, at least one of the transistors within the region has every terminal connected via a reconfigurable switch to other terminals of other transistor(s) within the region.

In the preferred embodiment, the arrangement of reconfigurable switches S1–S24 is such that a permutation of switch states provides a majority of meaningful circuit connections of transistors P1–N8 for a selected transistor arrangement, but so that there are less than the total number of possible transistor terminal connections. Thus, the transistors P1–N8 and the switches S1–S24 are arranged so that module **100** provides an extensive number of functional circuits for the permutation of switch S1–S24 states.

In one embodiment, a means to accomplish this is to arrange the transistors P1–N8 in layers between the power source terminal **10** and the power sink terminal **20**. The interconnections of terminals of the transistors P1–N8, including transistor to power terminal coupling, transistor to transistor coupling, and transistor self coupling generally allow non-conflicting current flow paths. In general, terminals should be coupled to facilitate current flow. In other words, a terminal that typically provides current should be coupled to at least one that typically receives current. For example, a drain terminal of a P-type transistor typically is coupled via a reconfigurable switch to the source terminal of another P-type transistor, such as S2 or S5, or to a drain terminal of an N-type, such as S3, S11, or S15. Likewise, the drain of a P-type transistor typically is coupled via switch to the drain of an N-type, such as S10, S11, S14, or S15. As such, transistor P1, P3, N5, and N7, along with transistors P2, P4, N6, and N8, provide current flow paths between the power source terminal **10** and the power sink terminal **20**.

Additional terminal interconnections may be used to provide additional meaningful terminal to terminal connectivity and functionality (for example, transistor bypass

switches, such as by S3, S7, S11, S12, S15, S20, or S22). Also, one or more transistor bias switches, ones connected from the gate to the drain of a transistor, such as by S4, S13, S16, or S23, can be provided to provide addition connectivity.

Moreover, like terminals of different transistors may also be coupled to provide addition meaningful connectivity. For example, one or more of the gate terminals of transistors may be connected together, such as by S1, S9, S17, or S24, or one or more of source or drain terminals may be connected together, such as by S8 or S18. It also is possible to provide additional meaningful connections by providing terminal connections between transistor terminals not otherwise directly coupled, such as by S6. The amount and type of interconnections allow evolution to explore and utilize a variety of basic analog and/or digital circuits, of various granularities such as, for example, current mirror, differential current pair, amplifier, multi-input NAND gate, multi-input XOR gate, inverter, and more, which may be formed through appropriate switch activation, to arrive at a solution.

The types of interconnections represented in FIG. 2 by switches S1–S24, may be used in embodiments of the present invention to provide the responsive means. Not all of the switches are necessary. Additionally, other connections not shown may be used. For example, an additional switch may be connected from the drain of P3 to the drain of N8, or across the source and drain of P4. On the other hand, in embodiments where switch state is controlled by serially shifting to each switch to download each bit of the chromosome string, more switch couplings will take more time to program and not necessarily provide additional benefits. In yet other embodiments, some of the transistor bypass switches, such as S12, S15, and S22 may be omitted to facilitate cascading of additional modules.

Furthermore, although preferred, it is not necessary with the present invention, to provide a module having eight transistors. The number of transistors maybe greater or less than eight. The optimal number and arrangement is task dependent. Eight transistors of complimentary conduction types, such as N-type and P-type, is expected to provide a universal application for an evolvable analog circuit. Likewise, pairing the transistors P1–P2, P3–P4, N5–N6, and N7–N8 within the module, as shown in FIG. 2, is not necessary in all embodiments of the present invention. Pairing of the transistors, however, is expected to provide a universal application for an evolvable analog circuit.

Similarly, arranging the transistors in a particular number of layers within the module, such as for example the four layers formed by the transistor pairs P1–P2, P3–P4, N5–N6, and N7–N8, as shown in FIG. 2, is not necessary in all embodiments of the present invention. Some preferred embodiments may have six, eight, or more layers. Fewer layers also is possible (e.g. three or two or possibly a single layer). Providing about four layers of the transistors, however, is expected to provide a universal application for an evolvable analog circuit. The layered structure of the preferred embodiment is patterned after some common designs of analog and digital structures. Other embodiments, however, may have a structure tailored for a specific application.

In some embodiments, the terminal interconnections are provided to facilitate power flow from power source 1.0 to power sink **20**, along with left to right lateral and/or laterally and downward connection. Such interconnection strategy, with a module output at, for example, P4 and/or N6, and/or others, allows for cascading signal flow to additional

modules, which may be added if desired to provide further evolutionary solutions. Further, feedback type interconnections are also possible in some embodiments, within a module, or more preferably between modules.

In the embodiment of FIG. 2, only four layers of pairs of transistors, two PMOS and two NMOS, were chosen for simplicity. The PTA architecture allows the implementation of bigger circuits by cascading PTA modules. Embodiments of the present invention may have each chip implementing one PTA module, or many modules may be present on each chip. To offer sufficient flexibility the module of this embodiment has all transistors terminals, except those connected to power and ground, connected via switches to expansion terminals, not shown, to allow switchable coupling of input and output signals to modules and to allow interconnection between modules to facilitate evolution.

Various module interconnections are possible. For example, some embodiments may have modules coupled so that a module is connected to adjacent modules, such as in north, south, east, and west to form a neighborhood, and have additional connectivity with one or more modules from a different neighborhood, such as jumping over a set of modules, for example jumping one to four modules, to a different neighborhood. It is possible to have one to four interconnections locally while having less extending to modules in other neighborhoods.

In one embodiment expected to provide a universal application for evolvable analog circuits, it is preferred to have higher density of module to module interconnections in a center portion of a layout and a lower density of interconnections at the periphery. This is because, in some situations, too many module to module interconnections could complicate evolution and be detrimental to stabilization. In other situations, more interconnections may be necessary to provide a more complicated solution. As this may not be established before array layout and fabrication, a non-homogeneous layout is expected to provide more flexibility when for evolutionary purposes.

In one possible embodiment, because it is not always known beforehand how many modules are needed to evolve a solution, it may be possible to use many internal test points on the PTA as possible outputs for evaluation of the distance of the response from the target response. This may be implemented using individuals in the same population with different sizes (chromosome length), or simply parallel evaluation of the routed outputs of many circuits. A part of the genetic code could indicate where the output is to be probed.

In some embodiments, the programmable or reconfigurable switches S1–S24 are implemented with transistors, such as a pair of NMOS-PMOS back to back transistors acting as a simple T-gate switch. In these embodiments, the transistor T-gate switches are selected: to pass analog signals; to have the resistance of each switch variable between low (approximately tens or hundreds of ohms) and high (in excess of tens and hundreds of Mohms and above); and to preferably provide an intermediate resistance (although its linearity is not necessarily important in these embodiments). In practice, the switches are non-ideal in that they have a big, but finite, resistance in the OFF state (approximately Mohms or Gohms) and a non-zero resistance/impedance in the ON state (approximately tens of Ohms). This feature can facilitate evolution. While the effects of non-ideal switches may be negligible in a first approximation for many digital circuits, such effects may fundamentally affect analog programmable circuits.

Turning to FIG. 3, besides leading to designs that are possible by human designers, evolution with non ideal components can lead to circuits that are unusual from the perspective of typical design practice. For example, synthesis of a DC circuit with a Gaussian response was performed in hardware in four separate chips. The four chips were programmed in parallel with bit-string configurations corresponding to four individuals of a population of 1000; then, tie next four were programmed, and so on until all 1000 in one generation were tested. Evolution led to "Gaussian" circuit solutions within 20–30 generations.

In this example, the genetic algorithm parameters in one of the runs are as follows: Population: 1000; Chromosome size: 24 bits (1 PTA); 52–88 bits (2PTAs, variable depending on interconnection schemes); Evaluation samples: 30; Mutation rate: 4%; Cross Over rate: 70%; Tournament Selection: 20 individuals; Elit Strategy: 9% population size (88 individuals); Fitness Function: Square root Mean Error; Fitness[indiv.]=target$(xi)^2$–indiv$(xi)^2$}.

FIG. 3 shows four different hardware-evolved mutant solutions for a Gaussian curve provided by modules 1–4. The mutations in the genetic code of the solutions obtained by evolution, depicted vertically as chromosomes R24–R1 that correspond to switches S24–S1 of FIG. 2, may be compared with the human-designed solution of the Gaussian circuit located to the right of R24–R1. While those observed mutants had 2–4 switches away of the same solution humans would design, it is expected this property can be exploited by evolution to ultimately provide quite different solutions, possibly better, than a human designer in other applications. Even if this is not the case, such mutants may increase the number of valid solutions, thus easing the search for a solution. Moreover, it may provide alternate, or backup solutions, that may be used in the event that an evolved solution becomes invalid due to component faults or other irregularity.

Turning to FIG. 4, other embodiments of the present invention may employ gray or gradual switches as the reconfigurable switches S1–S24 of FIG. 2 to introduce an annealing effect to the evolutionary process. Instead of ON/OFF, the switches were considered as having a resistance Low/High (Low for ON state). The binary genetic code would thus specify if the switch is Low or High, but the numerical meaning of this qualitative code (e.g. a resistance related parameter) would change gradually as a function of a temperature-like parameter as illustrated in FIG. 4.

Initially the temperature is high, and Low and High switch status have values close to each other, such as 2 Mohms for Low and 20 Mohms for High. Gradually, the temperature goes down and the switch resistance polarizes to the extremes of for example 10's of Ohms for Low and 10's of Gohms for High. The number of generations was chosen to ensure some quasi-static behavior. This is because the response of the same best individuals from older generation differed in the newer generation because the circuit had different resistance for switches.

This evolution through a gradual morphing process proved more efficient in simulation than searches in which the switches High or Low of extreme values, such as 10's of Ohms for Low and 10's of Gohms for High, were fixed at all times. Thus, evolution using gradual switch values provided as much as an order of magnitude faster solution.

Many solutions observed while running through this "freezing" process were acceptable solutions with switches partly open, in effect with all transistors taking part in generating the function. It may be preferred in some embodi-

ments to allow the annealing process to continue until the switch states are frozen or polarized to their full high and low impedance states. This provides more robust solutions, ones that are not as sensitive to thermal changes of transistors, or to other transistor dependent variations. In other embodiments, the switches may not be completely polarized. In such embodiments, the annealing process may be shortened, or not used at all, to provide solutions having switch states not completely polarized to their full high and low impedance states. Such solutions may be more desirable in certain instances, such as, for example, where design speed, resources, functionality, or other constraints make such solutions desirable.

In such situations where design speed is an important factor, it is possible to provide a fitness function which includes fitness gradient information to characterize the likelihood of success of individual during the morphing process. As such, individuals not showing sufficient improvement in fitness during the morphing process could be abandoned to expedite the evolution process.

Gradual switches allow variable control over the resistance of the switches so provide a means for variably controlling transistor coupling. The gradual switch may be implemented with transistors, such as a pair of NMOS-PMOS back to back transistors acting as a simple T-gate switch. The switches are selected as discussed above. In the implementation of one embodiment, the meaning of high or low is controlled by an A/D converter, either external or internal to the chip. In such an embodiment, a single bit of the bit string may be represented by several bits to provide adjustment of the value of high and low. Thus, a single bit used to qualitatively represent high or low is delivered to the A/D converter, which in turn provides a quantitative analog value of the high/low states to control the annealing process.

In one embodiment, to simplify device fabrication, the resistance value assigned the switches to represent high and low are controlled together to provide homogeneous control of high/low values of the switches. It also is possible in some embodiments to provide heterogeneous control of the switch annealing process to provide non-uniform, or even individual control of the values representing high and low.

A fitness function which considers shape information by using a weighted combination of parameters x1–x7 as depicted in FIG. 5 also can improve evolution search efficiency over a Euclidian type fitness function in some applications. Fitness functions, such as the one of FIG. 5 used lo evolve a Gaussian response circuit, however, generally do not have broad applicability.

While the preferred embodiments and methods of the present invention have been described in detail above, many changes to these embodiments and methods may be made without departing from the true scope and teachings of the present invention. The present invention, therefore, is limited only as claimed below and the equivalents thereof.

We claim is:

1. A method for evolving a circuit comprising:

a) configuring a plurality of transistors using a plurality of reconfigurable switches so that each of the plurality of transistors has a terminal coupled to a terminal of another of the plurality of transistors such that a single reconfigurable switch controls respective couplings therebetween; and

b) controlling the plurality of reconfigurable switches in response to a chromosome pattern.

2. claim 1 wherein controlling the plurality of reconfigurable switches further comprises using an annealing function.

3. claim 2 wherein controlling the plurality of reconfigurable switches comprises:

a) selecting qualitative values for the plurality of reconfigurable switches in response to the chromosomal pattern;

b) selecting initial quantitative values for the selected qualitative values; and

c) morphing the initial quantitative values.

4. claim 3 further comprising evaluating circuit fitness in response to morphed quantitative values.

5. claim 4 wherein morphing comprises selecting quantitative values more divergent than the initial quantitative values.

6. claim 5 wherein morphing comprises completely polarizing the qualitative values.

7. claim 5 wherein morphing comprises partially polarizing the qualitative values.

8. claim 5 wherein selecting initial quantitative values for the qualitative values comprises selecting quantitative values within two orders of magnitude.

9. claim 8 wherein selecting initial quantitative values for the qualitative values comprises selecting quantitative values within an order of magnitude.

10. claim 9 wherein morphing comprises completely polarizing the qualitative values.

11. claim 5 wherein morphing comprises providing a sufficient number of generations to ensure a gradual change in circuit response.

12. claim 2 wherein controlling the plurality of reconfigurable switches comprises homogeneously controlling the plurality of reconfigurable switches.

13. claim 2 wherein controlling the plurality of reconfigurable switches comprises heterogenously controlling the plurality of reconfigurable switches.

14. A method for evolving a circuit comprising:

a) coupling a plurality of transistors with a plurality of reconfigurable switches so that each of the plurality of transistors within a region of the circuit is coupled via a single reconfigurable switch to a terminal of another of the plurality of transistors; and

b) controlling the plurality of reconfigurable switches in response to a chromosome pattern.

15. claim 14 wherein controlling the plurality of reconfigurable switches further comprises using an annealing function.

16. claim 15 wherein controlling the plurality of reconfigurable switches comprises:

a) selecting qualitative values for the plurality of reconfigurable switches in response to the chromosomal pattern;

b) selecting initial quantitative values for the selected qualitative values; and

c) morphing the initial quantitative values.

17. claim 16 further comprising evaluating circuit fitness in response to morphed quantitative values.

18. claim 17 wherein morphing comprises selecting quantitative values more divergent than the initial quantitative values.

19. claim 18 wherein morphing comprises completely polarizing the qualitative values.

20. claim 18 wherein morphing comprises partially polarizing the qualitative values.

21. claim 18 wherein selecting initial quantitative values for the qualitative values comprises selecting quantitative values within two orders of magnitude.

22. claim 21 wherein selecting initial quantitative values for the qualitative values comprises selecting quantitative values within an order of magnitude.

23. claim **22** wherein morphing comprises completely polarizing the qualitative values.

24. claim **18** wherein morphing comprises providing a sufficient number of generations to ensure a gradual change in circuit response.

25. claim **15** wherein controlling the plurality of reconfigurable switches comprises homogeneously controlling the plurality of reconfigurable switches.

26. claim **15** wherein controlling the plurality of reconfigurable switches comprises heterogenously controlling the plurality of reconfigurable switches.

27. A method for evolving a circuit comprising:

   a) configuring a plurality of transistors between a power source and a power sink with a plurality of reconfigurable switches so that each of at least two terminals of each of the plurality of transistors within a region of the circuit are coupled via a single reconfigurable switch to a terminal of another of the plurality of transistors;

   b) selecting qualitative values for the plurality of reconfigurable switches in response to a chromosomal pattern;

   c) selecting initial quantitative values for the selected qualitative values;

   d) morphing the initial quantitative values; and

   e) evaluating a fitness of the circuit in response to morphed quantitative values of the chromosomal pattern.

28. claim **27** wherein morphing comprises completely polarizing the qualitative values.

29. claim **27** wherein morphing comprises partially polarizing the qualitative values.

30. claim **27** wherein selecting initial quantitative values for the qualitative values comprises selecting quantitative values within two orders of magnitude.

31. claim **30** wherein selecting initial quantitative values for the qualitative values comprises selecting quantitative values within an order of magnitude.

32. claim **31** wherein morphing comprises completely polarizing the qualitative values.

33. claim **27** wherein morphing comprises providing a sufficient number of generations.to ensure a gradual change in circuit response.

34. claim **27** wherein controlling the plurality of reconfigurable switches comprises homogeneously controlling the plurality of reconfigurable switches.

35. claim **27** wherein controlling the reconfigurable switches comprises heterogenously controlling the plurality of reconfigurable switches.

* * * * *